

REMARKS

In response to the Office Action mailed on Dec. 27, 2007, Applicant(s) respectfully request reconsideration.

Claims 1-17 and 20-52 are now pending in this Application.

In this Amendment, claims 1, 12, 13, 30, 33, 34, 49 and 51 been amended and claims 10, 11, 29, 33 and 50 have been cancelled.

Claims 1, 33, 34, 49 and 51 are independent claims and the remaining claims are dependent claims.

Applicant(s) believe that the claim(s) as presented are in condition for allowance. A notice to this affect is respectfully requested.

Claim 1, 14, 15, 21-23 and 49 have been rejected under 35 U.S.C. §102(e) as being anticipated by Silbershatz, et al. Claims 2-13, 16, 17, 20, 24, 25, 28-48 and 51 have been rejected under 35 U.S.C. 103(a) based on Silbershatz, et al. in view of Frank, et al., U.S. Pub. No. 2004/0250254 (Frank '254). Applicant(s) respectfully disagree(s) with these contentions and assert that the present claimed invention is not anticipated by any disclosure in the Silbershatz or Frank '254 references, alone or in combination.

Claim 50, rejected under 35 U.S.C. 101, has been herein cancelled; hence the rejection is deemed moot.

The previous response amended the manner in which process activation occurs, to which Frank was cited as teaching user controlled activation (context switching). Applicant further refines and distinguishes the distinctive activation provided by the claimed invention by clarifying that activation occurs with a genericizing reference that allows invocation of one of a class of events, while still allowing event specific parameters, now discussed in further below. Applicant submits that this amendment, clarifying subject matter already claimed, merely clarifies features of applicant's invention and does not present any new issues nor require a new search.

Frank does not show, teach, or disclose such a genericizing reference, as currently recited in claim 10. While Frank '254 appears to disclose classification

of event types, the claimed genericizing reference is transparent to event types, and may refer to any of the plurality of events, while each of the events in the class nonetheless employs event specific data for that particular event, as discussed further at page 11, lines 3-13. Thus, each of the events in an event class may employ the typeless genericizing reference for subsequent typecasting, in contrast to Frank which shows only an event-to-thread mapping [0094,0095]. While the Frank mapping includes a “catch all” for the mapping, this is merely a default when NO class applies, not a genericizing reference applicable to EACH class of events [0096-0099]. Accordingly, claim 1 has been herein amended with the subject matter of claim 10 to recite defining a plurality of events, the plurality of events associated with a genericizing reference, the genericizing reference inclusive of the plurality of events and each of the events associated with an event specific class having event data indicative of event specific parameters, to further clarify and distinguish applicant's claimed invention.

In further distinction of Applicant's claimed invention, the genericizing reference provides a bifurcation of event specific data and event class data, a further feature not shown or disclosed in Frank '254, alone or in combination. Accordingly, claim 1 has been further amended with the subject matter of claim 11, to recite that the event data including event variables generated and passed by the publisher of the event and subscriber instantiated variables generated by the state information of the subscriber. Accordingly, it is respectfully submitted that neither Frank '254 nor Silbershatz (previously cited) show, teach, or disclose, the genericizing reference and related claimed features.

In further distinction, as reflected in a telephone conference of April 24, 2008, between Examiner Nathen Price and Applicant's representative, Christopher J. Lutz, Frank '254 disclosed only a predetermined set of event types operable for signaling ([0091]). In contrast, Applicant's invention teaches a developer defined event that is independent of the underlying transport mechanism (i.e. code definitions) and delivery system (the developer need not be

constrained by predetermined event types). The event infrastructure transparently processes the subsequent publishing and invocation of the appropriate event handler within the subscriber. The event handler, corresponding to the event definition, performs remedial and responsive processing of the event. A developer need only develop entities to subscribe to an event to identify the appropriate handler, and publish the event, from the same or another module, with any related event data upon detection. Each particular event avoids extraneous event definitions and associated files because of the genericizing reference to the handler, thereby relieving the developer from external classes of unrelated events and allowing focus on the pertinent event class. A detecting software entity initiates event handling by publishing the event, upon which the event service identifies and invokes each subscriber entity for performing the event handler associated with the event, as discussed further at page 7, lines 21-29. Thus, each event is developer, or programmer defined independently of the event handler or delivery system underlying the event infrastructure. Accordingly, claim 1 has been herein amended to recite that each of the events [is] defined independently of an underlying delivery infrastructure, to further clarify and distinguish Applicant's claimed invention.

Claims 33, 34, 49, and 51 reciting at least similar limitations, have been likewise amended.

Claims 29, 32 and 50 have been deemed to be merely cumulative with the remaining claims and have therefore, in the interests of furthering the case to allowance, have been cancelled.

As the remaining claims depend from, either directly or indirectly, from claims 1 and 34 it is respectfully submitted that all claims are now in condition for allowance.

Applicant(s) hereby petition(s) for any extension of time which is required to maintain the pendency of this case. If there is a fee occasioned by this response, including an extension fee, that is not covered by an enclosed check, please charge any deficiency to Deposit Account No. 50-3735.

If the enclosed papers or fees are considered incomplete, the Patent Office is respectfully requested to contact the undersigned collect at (508) 616-9660, in Westborough, Massachusetts.

Respectfully submitted,

/CJL/

Christopher J. Lutz, Esq.
Attorney for Applicant(s)
Registration No.: 44,883
Chapin Intellectual Property Law, LLC
Westborough Office Park
1700 West Park Drive
Westborough, Massachusetts 01581
Telephone: (508) 616-9660
Facsimile: (508) 616-9661

Attorney Docket No.: EMC03-22(03111)

Dated: April 24, 2008